

IN THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the present application:

1. (Currently amended) A method of operating a file server, comprising the steps of:
 - receiving a network request at said file server from a client device, wherein the network request is part of a network session;
 - ~~recording a state at of said file server at the time of said receiving the request, said state including information regarding a persistent connection between said server and a client device transparently to said client device, after receiving the request and prior to an offlining of the file server;~~
 - determining whether a recovery of said file server after said offlining is to be accomplished by rebooting said file server or by takeover by another server;
 - restoring said state of said file server upon a reboot of said file server or upon a takeover by said other server, transparently to said client device; and
 - ~~attempting to continue continuing the network session between with said client device and said file server, wherein said step of recording state further comprises the step of determining whether recovery will be accomplished by rebooting the server or takeover by another server, wherein said steps of recording and restoring are transparent to said client device after restoring said state.~~

2. (Currently amended) The method of claim 1, wherein said step of receiving a network transmission request ~~also~~ includes the steps of
 - acknowledging receipt of said network request; and

processing said network request.

3. (Currently amended) The method of claim 1, wherein said step of recording state includes determining automatically whether the processing of ~~a~~the network request is at a point where said state can be reliably recorded.

4. (Currently amended) The method of claim 3, wherein said step of recording state occurs at points based on ~~the~~ progress of processing of ~~a~~the network request.

5. (Original) The method of claim 4, wherein said state is recorded to a non-volatile storage.

6. (Currently amended) The method of claim 1, wherein said step of recording state at said file server occurs as part of an elective reboot or elective takeover of a server, the method further comprising:

ignoring current network requests; and

processing all active network requests; ~~and,~~

prior to recording said state.

7. (Original) The method of claim 6, wherein all currently active requests are processed to completion.

8. (Currently amended) The method of claim 1, wherein said step of recording state further comprises the step of determining whether ~~said server~~ prior shutdown of said file server was elective or non-elective.

9. (Currently amended) The method of claim 8, wherein said step of determining whether ~~said server~~a prior shutdown is of said file server was elective or non-elective is a function of a flag value stored in ~~said~~a non-volatile storage.

10. (Original) The method of claim 9, wherein said flag value indicates said server shutdown was elective.

11. (Original) The method of claim 9, wherein said flag value indicates said server shutdown was non-elective.

12. (Canceled).

13. (Currently amended) The method of claim 1, wherein said step of determining whether recovery ~~will~~is to be accomplished by rebooting the server or takeover by another server is a function of ~~said~~a flag value stored in ~~said~~a non-volatile storage.

14. (Currently amended) The method of claim 13, wherein said flag value indicates said recovery ~~will~~is to be accomplished by rebooting the server.

15. (Currently amended) The method of claim 13, wherein said flag value indicates said recovery ~~will~~is to be accomplished by takeover by another server.

16. (Original) The method of claim 1, wherein said step of restoring state further comprises determining whether recovery is by reboot or takeover by another server.

17. (Currently amended) The method of claim 16, wherein said step of determining whether recovery is accomplished by reboot or takeover by another server is a function of said flag value stored in ~~said a~~ non-volatile storage.

18. (Currently amended) The method of claim 17, wherein said reboot comprises ~~the steps of:~~ rebooting ~~the server's~~ an operating system of the file server; and rebuilding in-memory data structures to the state prior to said reboot.

19. (Original) The method of claim 18, wherein said rebuilding in-memory data structures further comprises fetching the state stored in said non-volatile storage to rebuild said in-memory data structures.

20. (Original) The method of claim 17, wherein said takeover comprises fetching the state stored in the non-volatile storage and rebuilding said in-memory data structures in another server using said state.

21. (Currently amended) The method of claim 1, wherein said step of attempting to continue the network session further comprises ~~the step of processing the remaining~~ an uncompleted portion of the ~~uncompleted~~ request.

22. (Currently amended) An apparatus including:

means for receiving a CIFS request at a file server from a client device, wherein the CIFS request is part of a CIFS session; and

~~means for recording a state of said file server at the time of said receiving the request; said state including information regarding a persistent connection between said server and a client~~

device transparently to said client device, after receiving the request and prior to an offlining of the file server;

means for determining whether a recovery of said file server after said offlining is to be accomplished by rebooting said file server or by takeover by another server; and

on-reboot, means for restoring said file server to said state as recorded, upon a reboot of said file server or upon a takeover by said other server, transparently to said client device; and

means for attempting to continue continuing the CIFS session between with said client device and said file server, wherein said step of recording state further comprises the step of determining whether recovery will be accomplished by rebooting the server or takeover by another server after restoring said state.

23. (Currently amended) The apparatus of claim 22, wherein said means for receiving a CIFS request includes a means for acknowledging receipt of said CIFS request and a means for processing the request.

24. (Currently amended) The apparatus of claim 22, wherein said means for recording state includes a means ~~to determine~~ for determining automatically whether the processing of a CIFS request is at a point where said state can be reliably recorded.

25. (Currently amended) The apparatus of claim 24, wherein said ~~means for~~ recording state occurs at points based on the progress of processing of a CIFS request.

26. (Original) The apparatus of claim 25, wherein said state is recorded to a non-volatile storage.

27. (Currently amended) The apparatus of claim 22, wherein said ~~means for recording~~ said state at said file server occurs as part of an elective reboot or elective takeover of a server, the apparatus further comprising:
means for ignoring current CIFS requests; and
means for processing all active CIFS requests; ~~and~~
~~—means for recording state.~~

28. (Original) The apparatus of claim 27, wherein all currently active requests are processed to completion.

29. (Currently amended) The apparatus of claim 22, wherein said means for recording state further comprises ~~a~~ means for determining whether ~~said a~~ prior server shutdown was elective or non-elective.

30. (Currently amended) The apparatus of claim ~~27~~29, wherein said means for determining whether said prior server shutdown was elective or non-elective is a function of a flag value stored in said non-volatile storage.

31. (Currently amended) The apparatus of claim 30, wherein said flag value indicates said prior server shutdown was elective.

32. (Currently amended) The apparatus of claim 30, wherein said flag value indicates said prior server shutdown was non-elective.

33. (Canceled).

34. (Currently amended) The apparatus of claim 22, wherein said means for determining whether recovery ~~will~~is to be accomplished by rebooting the server or takeover by another server is a function of ~~said a~~ flag value stored in ~~said a~~ non-volatile storage.

35. (Currently amended) The apparatus of claim 34, wherein said flag value indicates said recovery ~~will~~is to be accomplished by rebooting the server.

36. (Currently amended) The apparatus of claim 34, wherein said flag value indicates said recovery ~~will~~is to be accomplished by takeover by another server.

37. (Canceled)

38. (Currently amended) The apparatus of claim 37, wherein said means for determining whether ~~said~~ recovery ~~is~~is to be by reboot or takeover by another server is a function of ~~said a~~ flag value stored in ~~said a~~ non-volatile storage.

39. (Currently amended) The apparatus of claim 38, wherein said reboot ~~further~~ comprises:
~~means for rebooting the server's~~ an operating system of the server; and
~~means for rebuilding in-memory data structures to the state prior to said reboot.~~

40. (Currently amended) The apparatus of claim 39, wherein said means for rebuilding in-memory data structures further comprises means for fetching the state stored in said non-volatile storage to rebuild said in-memory data structures.

41. (Currently amended) The apparatus of claim 38, wherein said takeover comprises means for fetching the state stored in said non-volatile storage and rebuilding said in-memory data structures in another server using said state.

42. (Currently amended) The apparatus of claim 22, wherein said means for attempting to continue the CIFS session further comprises a means for processing the remaining portion of the uncompleted request.

43-53. (canceled)

54. (New) The method of claim 1, wherein the network session is a CIFS session.

55. (New) A method of operating a storage server, the method comprising:

receiving a CIFS request at said storage server from a client device during a CIFS session;
determining whether an elective offlining process of said storage server has been initiated;
in response to determining that an elective offlining process of said storage server has been initiated, then prior to an elective offlining of said storage server,
ignoring further incoming CIFS requests from said client device,
completing currently active CIFS requests,
determining whether a recovery of said storage server after said elective offlining is to be accomplished by rebooting said storage server or by takeover by another server, and
recording a state of said storage server in a non-volatile storage medium transparently to said client device, in a manner which depends on whether said recovery is to be accomplished by rebooting said storage server or by takeover by another server;
restoring said state of said storage server from said non-volatile storage medium during said recovery, transparently to said client device; and

continuing the CIFS session with said client device after restoring said state.

56. (New) The method of claim 55, wherein said recording state comprises determining automatically whether processing of the CIFS request is at a point where said state can be reliably recorded.

57. (New) The method of claim 56, wherein said recording state occurs at points based on progress of processing of the CIFS request.

58. (New) The method of claim 55, wherein said determining whether recovery is to be accomplished by rebooting said server or takeover by another server is a function of a flag value stored in the non-volatile storage medium.